

Unsupervised searches for cosmological parity violation: An investigation with convolutional neural networks

Peter L. Taylor^{1,2,3,*}, Matthew Craigie⁴, and Yuan-Sen Ting^{5,6,3,1}

¹*Center for Cosmology and AstroParticle Physics (CCAPP), The Ohio State University, Columbus, Ohio 43210, USA*

²*Department of Physics, The Ohio State University, Columbus, Ohio 43210, USA*

³*Department of Astronomy, The Ohio State University, Columbus, Ohio 43210, USA*

⁴*School of Mathematics and Physics, The University of Queensland, Queensland 4072, Australia*

⁵*Research School of Astronomy and Astrophysics, Australian National University, Cotter Road, Weston, Australian Capital Territory 2611, Australia*

⁶*School of Computing, Australian National University, Acton, Australian Capital Territory 2601, Australia*

 (Received 18 December 2023; accepted 28 March 2024; published 15 April 2024)

Recent measurements of the 4-point correlation functions (4PCF) from spectroscopic surveys provide evidence for parity violations in the large-scale structure of the Universe. If physical in origin, this could point to exotic physics during the epoch of inflation. However, searching for parity violations in the 4PCF signal relies on a large suite of simulations to perform a rank test, or an accurate model of the 4PCF covariance to claim a detection, and this approach is incapable of extracting parity information from the higher-order N -point functions. In this work we present an unsupervised method which overcomes these issues, before demonstrating the approach is capable of detecting parity violations in a few toy models using convolutional neural networks. This technique is complementary to the 4-point method and could be used to discover parity violations in several upcoming surveys including DESI, Euclid, and Roman.

DOI: [10.1103/PhysRevD.109.083518](https://doi.org/10.1103/PhysRevD.109.083518)

I. INTRODUCTION

A detection of parity violation in the large scale structure of the Universe would break the standard cosmological model. Such a measurement could provide crucial insights into the physics of the early Universe [1–7] or exotic physics [8–15], and parity violations may even be responsible for the observed matter-antimatter asymmetry [16].

Searches for parity violations have concentrated on the polarized cosmic microwave background (CMB) signal [8,17–21], gravitational waves [22–25] and most recently the clustering of galaxies in the late Universe [26–28], which is the primary focus of this work.

Several of these studies have found tantalizing hints of parity violation. Two recent CMB EB -measurements report detections at the 2.4σ [19] and 3.6σ -level [29], although it has been suggested this could be due to dust emission (see e.g. the discussions in [30–32]).

More recently two studies have reported detections of parity violations in the Baryon Oscillation Spectroscopic Survey (BOSS) [33] CMASS sample at the 2.9σ [28] (hereafter Ph22) and 7.1σ [27] (hereafter Hou22). These detections are made from measurements of the 4PCF which are enabled by recent computational advances [34–37].

If physical in origin, the cause of this signal must also be consistent with the null detection of parity violation in the temperature trispectrum of the CMB [20].

In three dimensions the 4PCF is the lowest order N -point correlation function which is sensitive to parity violations [26]. To see this, it is useful to refer to Fig 1 where galaxies are treated as vertices of a tetrahedron. After defining a notion of the tetrahedra handedness about a prespecified primary vertex, we see that mirroring a tetrahedron about a vertex, which is equivalent to a parity transformation up to a rotation, switches the handedness of the tetrahedron. This is in contrast to a triangle or a line segment, where a reflection about a point is equivalent to a three-dimensional rotation.

The 4PCF approach has three main drawbacks:

- (i) First, even in the absence of observational systematics, quantifying the statistical significance is theoretically and computationally demanding. Hou22 quantified the uncertainty by computing the covariance of the 4PCF. The covariance has both a large number of terms and potentially poorly understood 8-point function contributions. This approach also assumes that the parity odd 4-point function is Gaussian distributed. Meanwhile, Ph22 performed a rank test and compared the level of parity violation in the data to a suite of 2048 MultiDark-Patchy mock realizations [38,39] of the CMASS footprint.

*taylor.4264@osu.edu

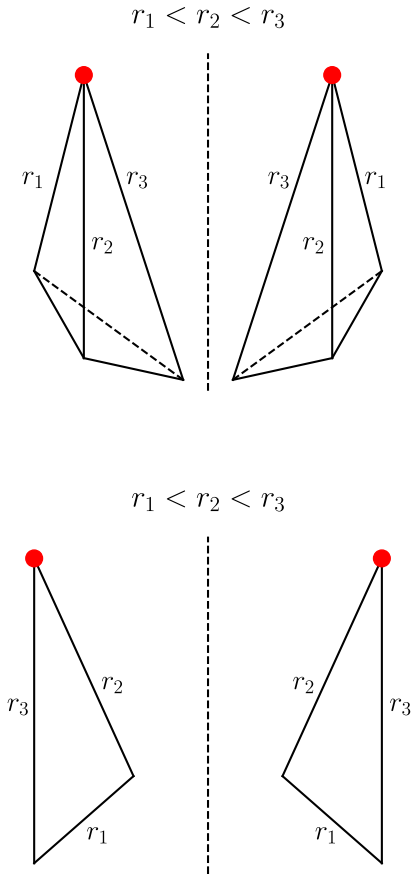


FIG. 1. In three dimensions, a tetrahedron is the lowest order polyhedron which is sensitive to parity i.e., handedness. Top: for the purposes of this work we say a tetrahedron is right-handed if when moving the right hand to the primary vertex with the thumb pointed into the tetrahedron, the fingers curl from the shortest to longest edges connected to the primary. It is impossible to rotate a left-handed tetrahedron (left) onto its mirror image (right), which is equivalent to a parity transform up to a 180° rotation. Bottom: in contrast the left-handed triangle (left) is equivalent to rotating the right-handed triangle (right) on the right “out of the page.”

These simulations have not been extensively verified at the 4-point level, and may lead to an underestimate of the uncertainty.

- (ii) Second, while it is in principle possible to extract parity information from the higher order N -point functions [40], this quickly becomes intractable for larger N . This potentially leaves a large-part of the parity odd signal untapped.
- (iii) Since the 4PCF is a global measure, it is difficult to localize which physical regions contribute most to the parity asymmetry. Ideally one could identify the regions where parity violations are strongest and perhaps even cross-correlate with known systematics to determine whether systematics are responsible for observed parity violations.

To address these shortcomings, we propose an unsupervised method which can in principle detect parity violation

using information from all N -point functions *from the data alone*. This technique may be employed to search for parity violations in large spectroscopic surveys including Euclid¹ [41,42], the Nancy Grace Roman Space Telescope² [43], the Dark Energy Spectroscopic Instrument (DESI)³ [44].

Intuitively, the algorithm described in this paper is simple. The first step is to divide the large-scale structure map into a large number of subvolumes. Then, for each subvolume, one can parity flip the volume before passing both the original and parity flipped subvolumes through a convolutional neural network (CNN). If the CNN can reliably classify which is the true realization and which has been parity flipped, this implies that the data is not parity invariant.⁴

Similar techniques have been employed to determine whether a photograph has been mirrored [45], distinguish images of mirrors from transparent glass [46] and classify chirality in medical imaging [47]. In high-energy physics, a similar technique has been developed to search the Large Hadron Collider’s calorimeter data for evidence of parity violation [48,49]. It is the technique of these former two works which this paper follows most closely. While we focus exclusively on CNNs in this paper, the method is generalizable to any architecture. In a follow up paper [50], we also consider neural fields and scattering networks and demonstrate that these can significantly outperform CNNs.

The unsupervised parity detection algorithm is described in Sec. II. It is then applied to 2D and 3D toy models in Sec. II and III. Finally in Sec. IV we conclude, by outlining a road map for future work.

II. UNSUPERVISED SEARCHES FOR PARITY VIOLATIONS USING CONVOLUTIONAL NEURAL NETWORKS

A. Algorithm

The algorithm described below and summarized in Fig. 2 was originally proposed in [48] (hereafter Le21) in the context of particle physics.

Consider a CNN, g , which maps some input, x , to $g(x) \in \mathbb{R}$. In the context of this work, one should think of x as a subvolume of a pixelized (or voxelized) galaxy overdensity map. Let us also define a parity operator, P . In two dimensions, P is equivalent to mirror symmetry, while in three-dimensions the parity operator flips a volume along all axes.

Intuitively, to detect a parity violation we would like the network, g , to maximize the distance between $g(x)$ and $g(Px)$. To build such a network, define a new function

¹<http://euclid-ec.org>.

²<https://www.nasa.gov/roman>.

³<https://www.desi.lbl.gov/>.

⁴In practice, the problem is recast as a maximization problem rather than a classification problem. See Sec. II A.

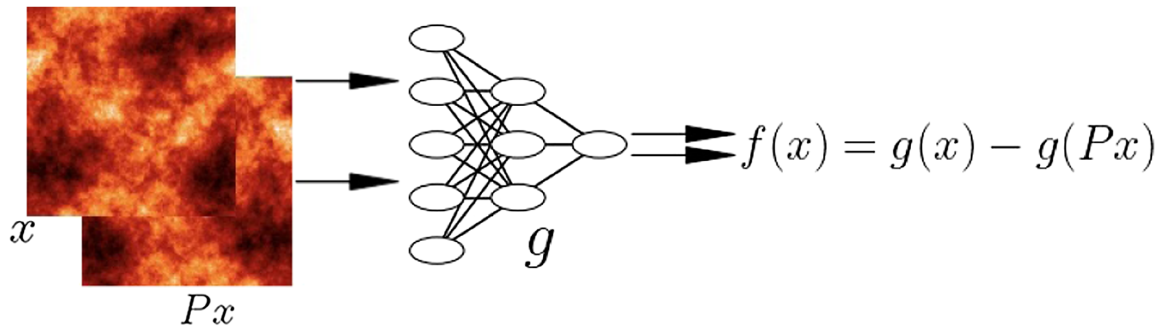


FIG. 2. Schematic showing the algorithm presented in Sec. II A. Some data, x , and the parity flipped data, Px , are both passed through a CNN denoted, g , such that $g(x) \in \mathbb{R}$. For every pair we define, $f(x) = g(x) - g(Px)$. The network learns to maximize the difference between $g(x)$ and $g(Px)$ during training (see Sec. II A for more details). As shown in Sec. II B, the expected value of $f(x)$ is exactly zero if no parity violations are present. Any deviation from zero is indicative of parity violation, while the statistical significance can be estimated from bootstrapping over the validation set.

$$f(x) = g(x) - g(Px). \quad (1)$$

Then for a batch, $B = \{x_1, x_2, \dots, x_N\}$, one might naïvely expect that choosing a loss function to maximize the mean output of the batch would achieve the objective. However, trivially rescaling all the weights and biases of the network by some constant $\alpha > 1$ achieves a smaller loss, so instead we define the loss-function,

$$-\mathcal{L}_B = \mu_B / \sigma_B, \quad (2)$$

where the mean, μ_B , and variance, σ_B , over the batch are defined as

$$\mu_B = \frac{1}{N} \sum_{x \in B} f(x), \quad (3)$$

and

$$\sigma_B^2 = \frac{1}{N} \left(\sum_{x \in B} [f(x)]^2 \right) - \mu_B^2. \quad (4)$$

After splitting the data into training and validation sets, the network can be trained in the usual fashion before using the validation set to detect parity violations. We describe this procedure in detail in the following section.

B. Network behavior, robustness to overtraining and parity violation detection criteria

Given the choice of loss in Eq. (2), the network learns to return a maximal positive number with small variance over the training set. Thus, if the mean value of $f(x)$ over the validation set, $\langle f(x) \rangle_{x \in V}$ (hereafter written $\langle f \rangle_V$), is greater than zero, then we have detected parity violation and the statistical significance can be quantified by bootstrapping over the validation set. Alternatively one could use $\langle f \rangle_V$ as

an extremely compressed parity odd summary and perform a rank test against a large number of simulations as in Ph22.

Our technique does not make false detections even when the network is overtrained. To see this it is useful to consider the case that no parity violations are present in the data. Then the probability that any given x is in the validation set is the same as the probability that Px is in the validation set. This observation has an important consequence. Let us suppose that after training $f(x) = n$, for some positive number n , then by design, $f(Px) = -n$. This implies that if no parity violations are present in the data, the expected mean value of $f(x)$ over the validation set, $\langle f \rangle_V$, is precisely zero. As noted in [48], this is true *even if the network is overtrained*.

It is worth noting that in order for the bootstrap uncertainties to remain valid, we implicitly assume that there is no covariance between the subvolumes in the validation set. In practice this can likely be achieved by choosing validation set subvolumes that are separated by large physical separations in the survey. This must be explicitly checked when working with real data. In [50], we also explicitly confirm that the distribution of the bootstrap resampled means is the same as the likelihood computed from simulations of independent universes.

III. RESULTS

In this section, we showcase how the method described in Sec. II A can be used to search for parity violations in the context of galaxy surveys. In this paper we use a parity-violating toy model described in Sec. III A. We show how the unsupervised technique is capable of detecting parity violations in these fields. We start by building intuition in 2D (see Sec. III B) before demonstrating that the technique extends to 3D (see Sec. III B). All CNNs are implemented in TensorFlow [51] Keras [52].⁵

⁵PyTorch [53] was also used during the development phase.

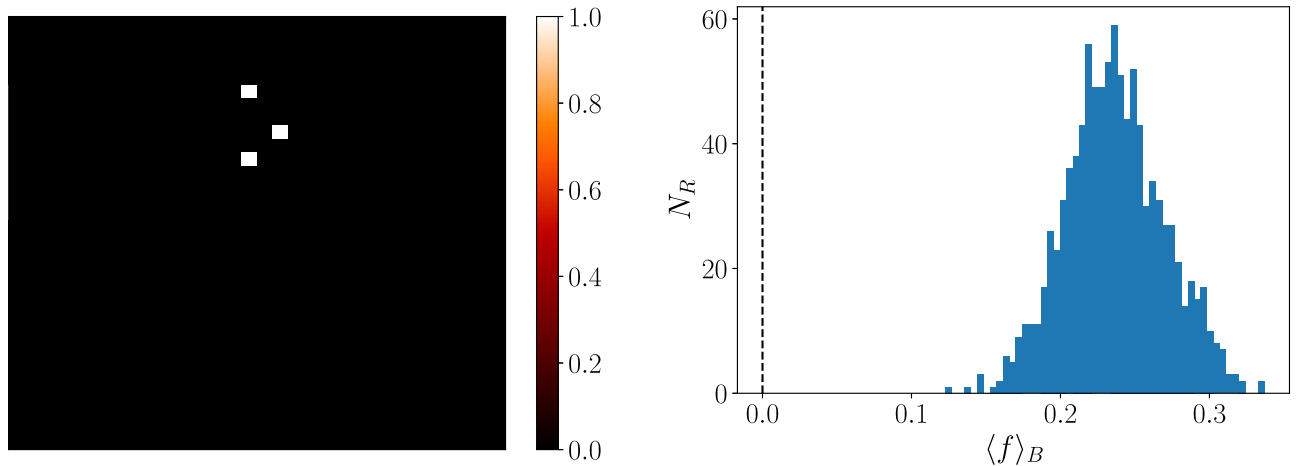


FIG. 3. Left: in this example, parity-violating “fields” are generated by mapping a fixed right-handed triangle, T , to a random location in the field and assigning a galaxy to each vertex (see Sec. III B 1 for more details). Right: histogram of the mean of the validation set over bootstrap resamplings, $\langle f \rangle_B$, while N_R gives the number of occurrences in the resampling in each histogram bin. This is greater than zero (indicated by a dashed line) at high-confidence ($>5\sigma$), so we have detected parity violations.

A. A toy model with parity and mirror asymmetries

To generate a 2D mirror asymmetric field, we:

- (1) start with a left-handed⁶ triangle (see Fig. 1 for more details) with a vertex placed at the origin,
- (2) apply a random isometry to the triangle in the plane,
- (3) place a galaxy inside each voxel containing a vertex of the triangle,
- (4) and repeat steps 1–3 N times to generate a galaxy field with $3N$ galaxies.

In 3D, rather than triangles we use right and left-handed tetrahedra. More complex parity-violating fields can be built by mixing right and left-handed tetrahedra or triangles (see e.g. Sec. III B 2).

B. Toy models in two dimensions

1. 2D model 1: 1 triangle per field

Following the procedure described in Sec. III A, we use the triangle, T , with vertices in the plane at $(0, 0)$, $(4, 0)$ and $(0, 8)$ to generate a set of “random fields.” We apply a random isometry by applying a random rotation, $\theta \in [0, 2\pi]$, and uniformly drawing a translation, $\Delta x, \Delta y \in [-23, 23]$ inside each field.⁷ The field is then pixelized on a 32×32 grid centred on the origin. A randomly chosen realization is shown on the left of Fig. 3.

We produce a training set of 10^4 realizations and a validation set of 10^3 realizations before training a network to detect parity violations following the procedure outlined in Sec. II A. A full description of the network architecture and training is given in Appendix A 1.

⁶It is equally valid to use a right-handed triangle instead.

⁷This choice ensures that all the vertices of the triangle remain inside the field.

An example of the training data is shown on the left hand side of Fig. 3. Meanwhile a histogram of the mean of the validation set over bootstrap samples is shown on the right hand side. This is greater than zero at every resampling so we have made a clear detection of mirror asymmetry. Even by eye, one can easily “detect” mirror asymmetries, so we now consider a more challenging example.

2. 2D model 2: Complex triangle example

We use the triangle, T , defined in the previous example and the mirror image, PT . In each field we draw N_T copies of triangle T and N_{PT} copies of triangle PT where N_T and N_{PT} are respectively drawn uniformly from the discrete sets $\{31, 32, 33, 34, 35, 36, 37\}$ and $\{42, 41, 40, 39, 38, 37, 36\}$ inducing a small amount of parity violation. Given all the possible triangles that can be formed from these vertices, this represents less than a 1% relative overabundance of right-handed triangles on average. We then apply an independent isometry to each galaxy as in the previous example, before placing a galaxy at each vertex and pixelizing. Hence, on average there are 72.5 triangles per field, but with a slightly different amount of mirror asymmetry per field.

We generate 10^5 training realizations and 10^4 realizations for validation so that there are $\sim 7.5 \times 10^6$ “galaxies” in total. This is similar to the expected number of galaxies in upcoming spectroscopic surveys including Euclid, DESI and Roman.

A randomly chosen example of one of the fields generated following this procedure is shown on the top left of Fig. 4. Unlike the example in the previous section, it is impossible to detect parity violations by eye.

To search for parity violations, we train the network described in Appendix A 2. A histogram of the mean over

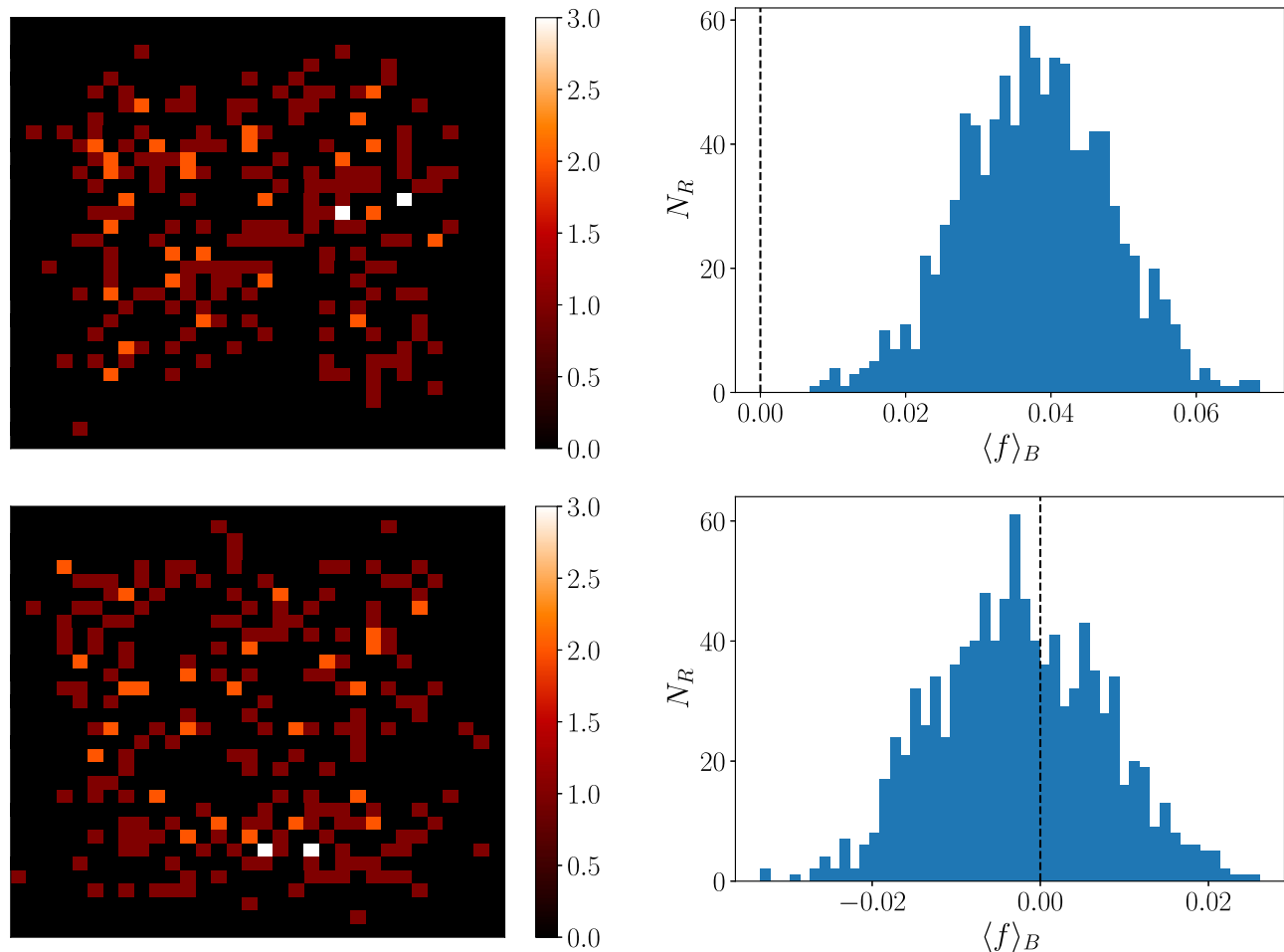


FIG. 4. Top left: an example of a parity-violating “field” generated by mapping left-handed and right-handed triangles (with an overabundance of one kind) to a random location in the field and placing a galaxy at each vertex. The fields have on average 72.5 triangles per field and we place on average 5 more right-handed compared to left-handed triangles per field. Given all the possible triangles that can be formed from these vertices, this represents less than a 1% relative overabundance of right-handed triangles on average. See Sec. III B 2 for more details. Top right: histogram of the mean of f from bootstrap resampling the validation set. Parity-violations are detected at the 3.9σ level. Bottom left: same as top left, but generated as a null test using a procedure where there is no overabundance of left versus right handed triangles (see Sec. III B 2) for more details). Bottom right: same as above. No parity violations are detected in the data as $\langle f \rangle_B$ is consistent with zero.

bootstrap resamples of the validation set displayed in the top right of Fig. 4. We clearly detect parity violations at the 3.9σ level.

As a null test, we repeat the experiment described above, but this time in each field we draw N_T copies of triangle T and N_{PT} copies of triangle PT from the same discrete set: $\{33, 34, 35, 36, 37, 38, 39\}$. With this choice, we expect 72 galaxies per field on average with no overabundance of right-handed or left-handed triangles.

A randomly chosen example of such a field is shown on the bottom left of Fig. 4. It is virtually indistinguishable from the parity-violating field in the top panel.

We generate a null training set of 10^5 training realizations and 10^4 validation realizations before training a network with the same architecture as before. The results are shown on the bottom right of Fig. 4. As expected, the

network does not detect parity violations when they are not present.

C. Toy models in three dimensions

1. 3D model 1: 1 tetrahedron per field

In this example we use a tetrahedron, T , with vertices at $(1, 0, 0)$, $(-3, 3, 0)$, $(-5, -5, -5)$ and $(0, 0, 3)$. Following the procedure outlined in III A, we apply a random rotation defined by 3 Euler angles and a translation $\Delta x, \Delta y, \Delta z \in [-10, 10]$. Each vertex is then mapped onto a $28 \times 28 \times 28$ -grid before a galaxy is placed inside each pixel containing a vertex.

We generate 10^4 training realizations and 10^3 validation realizations before training the network described in Sec. A 3 to search for parity violations. We are able to

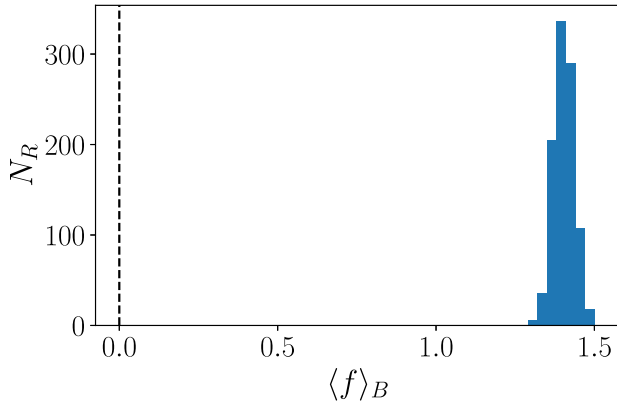


FIG. 5. Same as the right hand side of Fig. 3 but for this example we use a single right-handed tetrahedron per field. We make a clear detection ($>5\sigma$) of parity violation.

make a clean detection ($>5\sigma$) of parity violation as shown in Fig. 5.

The purpose of this exercise has been to demonstrate that we can in principle detect parity violations in 3D. However, in contrast to the 2D case, we find that the CNN fails to extract parity information with a more realistic number of galaxies in each field. In particular given the large number of free parameters in the model, we find that the CNN is prone to overtraining and the model does not generalize well to the validation set, which we use for detection. Convolutions in 3D are also prohibitively expensive and inefficient as much of the data volume contains empty pixels. For these reasons we will need to find a more efficient architecture before attempting to detect parity violations with real data. This is the subject of a follow-up paper [50].

IV. CONCLUSION

This paper presents a method to perform an unsupervised search for parity violations from spectroscopic galaxy surveys. This follows possible detections using the 4PCF method in Ph22 and Hou22. In principle a similar technique could be used to search for EB parity violations in weak lensing (see e.g. [54]) or CMB data.

We have demonstrated that CNNs are readily capable of detecting small amounts of mirror asymmetry in two-dimensions, but find that this result does not easily generalize to three dimensions. In a follow-up paper, we investigate deep scattering networks and neural fields and show they outperform the CNNs presented in this work. In the future we will explore architectures which are known to perform well on sparse three dimensional data [55], and point clouds [56–59] which have already been shown to perform well in the context of galaxy clustering [60–62].

Compared to 4PCFs, the unsupervised approach can be trained from the data alone and is not limited by the

number and accuracy of available simulations, nor does the detection significance rely on an accurate model of 8-point covariance contributions. The method is also in principle capable of extracting information from the higher-order N -point configurations, which quickly becomes computationally infeasible for N -point correlation functions at higher order. Furthermore one can make a map of f over the survey volume to determine whether the parity-violating signal is spatially correlated with known systematics.

Despite these advantages, the unsupervised approach has several drawbacks. First, it is not possible to fit a parity-violating physical model to constrain parameters of interest. Second, as with many machine learning algorithms the method suffers from a lack of interpretability. For these reasons, the unsupervised and 4PCF approach should be thought of as complementary.

The next step is to find an architecture which is capable of detecting parity violations in simulations where we know the signal is present e.g. [63,64]. We will then search for parity violations in the CMASS sample to verify the results of Ph22 and Hou22. A detection would provide the first confirmation using an independent technique.

The unsupervised approach presented in this work may become a powerful tool in the search for parity violations in the large-scale structure of the Universe. The next generation of spectroscopic surveys will prove an important testing ground. Euclid, Roman and DESI will employ very different survey strategies, so parity-violating observational systematics are unlikely to be shared across all three datasets. Thus, confirmation of parity violations using the methods presented in this paper in all three next generation datasets would put the existence of physical parity violations on firmer ground.

ACKNOWLEDGMENTS

P. L. T. is supported in part by NASA ROSES 21-ATP21-0050. This work received support from the U.S. Department of Energy under Contract No. DE-SC0011726. Y. S. T. acknowledges financial support from the Australian Research Council through DECRA Fellowship No. DE220101520. The authors thank Ashley Ross, Chun-Hao To, David Weinberg, Eric Huff, Zachary Slepian, Oliver Philcox, Erik Zaborowski, Maja Joblonska, Eduardo Roza, Tomasz Róžański, Iona Ciucă, Biprateep Dey, Jiamin Hou and Robert Cahn for enlightening conversations, and the anonymous referee for reviewing this paper. This work used public software [65–67]. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility located at Lawrence Berkeley National Laboratory, operated under Contract No. DE-AC02-05CH11231.

APPENDIX: NETWORK ARCHITECTURES

The network architectures used in Sec. III are summarized in this section. The networks are optimized up to the point where they are capable of detecting parity violations, but we make no claims that these architectures are optimal.

1. 2D model 1 architecture

For g , we take a fairly common network architecture using ReLu [68] activation functions with a single convolution layers of 10 filters with filter size of (10, 10) pixels, followed by a max pool layer of size (2, 2) with no strides. This is connected to a dense layer with 10 neurons before the final layer outputs a single value, $g(x)$. The network is trained for 10 epochs, using the Adam optimizer [69] with a constant learning rate of 10^{-3} and a batch size of 64. The learning rate and batch size are fixed for every network in this paper.

2. 2D model 2 architecture

The network architecture used in this example is loosely based on AlexNet [70]. The network starts with a convolution layer of 100 filters of size (11, 11) padded so that

input and output size remains unchanged, followed by a max pool layer of size (2, 2) with no strides. This is followed by two convolution layers with 100 filters of size (5, 5) padded so that input and output size remains unchanged, then three convolution layers with 100 filters of size (3, 3), followed by a max pool layer of size (2, 2) with no strides. This is then sequentially passed to two dense layers with 4000 neurons with a 50% dropout rate before being passed to a dense layer with 1000 neurons and before outputting $g(x)$. ReLu activation functions are used throughout. The network is trained for 10 epochs.

3. 3D model 1 architecture

The network starts with a 3D convolution layer with 100 filters of size 11 padded so that the size of the training data remains the same. This is followed by a max pool layer of size 2 with a stride length of 1. This is followed by four 3D convolutions of 100 filter of size 3 before being passed to a max pool layers 2 with a stride length of 1. After flattening, this is followed by 3 dense layers of size 4000, 1000 and 200 with a 50% dropout rate between each before outputting $g(x)$. ReLu activation functions are used throughout. The network is trained for 10 epochs.

-
- [1] D. Grasso and H. R. Rubinstein, *Phys. Rep.* **348**, 163 (2001).
 [2] P. A. R. Ade *et al.* (Planck Collaboration), *Astron. Astrophys.* **594**, A19 (2016).
 [3] M. Shiraishi, *J. Cosmol. Astropart. Phys.* **06** (2012) 015.
 [4] J. L. Cook and L. Sorbo, *Phys. Rev. D* **85**, 023534 (2012); **86**, 069901(E) (2012).
 [5] P. Adshead, J. T. Giblin, T. R. Scully, and E. I. Sfakianakis, *J. Cosmol. Astropart. Phys.* **12** (2015) 034.
 [6] G. Cabass, M. M. Ivanov, and O. H. E. Philcox, *Phys. Rev. D* **107**, 023523 (2023).
 [7] S. Jazayeri, S. Renaux-Petel, X. Tong, D. Werth, and Y. Zhu, *Phys. Rev. D* **108**, 123523 (2023).
 [8] A. Lue, L.-M. Wang, and M. Kamionkowski, *Phys. Rev. Lett.* **83**, 1506 (1999).
 [9] N. Bartolo and G. Orlando, *J. Cosmol. Astropart. Phys.* **07** (2017) 034.
 [10] S. Alexander and N. Yunes, *Phys. Rep.* **480**, 1 (2009).
 [11] N. Bartolo, S. Matarrese, M. Peloso, and M. Shiraishi, *J. Cosmol. Astropart. Phys.* **01** (2015) 027.
 [12] L. Bordin and G. Cabass, *J. Cosmol. Astropart. Phys.* **07** (2020) 014.
 [13] L. Pogosian and M. Wyman, *Phys. Rev. D* **77**, 083509 (2008).
 [14] I. Y. Rybak and L. Sousa, *Phys. Rev. D* **104**, 023507 (2021).
 [15] O. Özsoy, *Phys. Rev. D* **104**, 123523 (2021).
 [16] F. Schmidt, N. E. Chisari, and C. Dvorkin, *J. Cosmol. Astropart. Phys.* **10** (2015) 032.
 [17] M. Kamionkowski and T. Souradeep, *Phys. Rev. D* **83**, 027301 (2011).
 [18] M. Shiraishi, D. Nitta, S. Yokoyama, K. Ichiki, and K. Takahashi, *Prog. Theor. Phys.* **125**, 795 (2011).
 [19] Y. Minami and E. Komatsu, *Phys. Rev. Lett.* **125**, 221301 (2020).
 [20] O. H. E. Philcox, *Phys. Rev. Lett.* **131**, 181001 (2023).
 [21] O. H. E. Philcox and M. Shiraishi, *arXiv:2308.03831*.
 [22] N. Yunes, R. O’Shaughnessy, B. J. Owen, and S. Alexander, *Phys. Rev. D* **82**, 064017 (2010).
 [23] A. Wang, Q. Wu, W. Zhao, and T. Zhu, *Phys. Rev. D* **87**, 103512 (2013).
 [24] T. Zhu, W. Zhao, Y. Huang, A. Wang, and Q. Wu, *Phys. Rev. D* **88**, 063508 (2013).
 [25] A. Nishizawa and T. Kobayashi, *Phys. Rev. D* **98**, 124018 (2018).
 [26] R. N. Cahn, Z. Slepian, and J. Hou, *Phys. Rev. Lett.* **130**, 201002 (2023).
 [27] J. Hou, Z. Slepian, and R. N. Cahn, *arXiv:2206.03625*.
 [28] O. H. E. Philcox, *Phys. Rev. D* **106**, 063501 (2022).
 [29] J. R. Eskilt and E. Komatsu, *Phys. Rev. D* **106**, 063503 (2022).
 [30] S. E. Clark, C.-G. Kim, J. C. Hill, and B. S. Hensley, *Astrophys. J.* **919**, 53 (2021).
 [31] J. R. Eskilt, *Astron. Astrophys.* **662**, A10 (2022).
 [32] P. Diego-Palazuelos *et al.*, *Phys. Rev. Lett.* **128**, 091302 (2022).

- [33] S. Alam, F. D. Albareti, C. A. Prieto, F. Anders, S. F. Anderson, T. Anderton, B. H. Andrews, E. Armengaud, É. Aubourg, S. Bailey *et al.*, *Mon. Not. R. Astron. Soc.* **522**, 5701 (2023).
- [34] O. H. E. Philcox, J. Hou, and Z. Slepian, arXiv:2108.01670.
- [35] Z. Slepian and D. J. Eisenstein, *Mon. Not. R. Astron. Soc.* **454**, 4142 (2015).
- [36] Z. Slepian and D. J. Eisenstein, *Mon. Not. R. Astron. Soc.* **455**, L31 (2016).
- [37] O. H. E. Philcox, Z. Slepian, J. Hou, C. Warner, R. N. Cahn, and D. J. Eisenstein, *Mon. Not. R. Astron. Soc.* **509**, 2457 (2021).
- [38] F.-S. Kitaura *et al.*, *Mon. Not. R. Astron. Soc.* **456**, 4156 (2016).
- [39] S. A. Rodríguez-Torres *et al.*, *Mon. Not. R. Astron. Soc.* **460**, 1173 (2016).
- [40] R. N. Cahn and Z. Slepian, *J. Phys. A* **56**, 325204 (2023).
- [41] A. Blanchard *et al.* (Euclid Collaboration), *Astron. Astrophys.* **642**, A191 (2020).
- [42] R. Laureijs *et al.* (Euclid Collaboration), arXiv:1110.3193.
- [43] D. Spergel, N. Gehrels, C. Baltay, D. Bennett, J. Breckinridge, M. Donahue, A. Dressler, B. Gaudi, T. Greene, O. Guyon *et al.*, arXiv:1503.03757.
- [44] A. Aghamousa *et al.* (DESI Collaboration), arXiv:1611.00036.
- [45] Z. Lin, J. Sun, A. Davis, and N. Snavely, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2020), pp. 12295–12303, https://openaccess.thecvf.com/content_CVPR_2020/papers/Lin_Visual_Chirality_CVPR_2020_paper.pdf.
- [46] H. Tamura, K. E. Prokott, and R. W. Fleming, *J. Vis.* **21**, 14 (2021).
- [47] T. S. Kang, B. J. Kim, K. Y. Nam, S. Lee, K. Kim, W.-s. Lee, J. Kim, and Y. S. Han, *Sci. Rep.* **12**, 1444 (2022).
- [48] C. G. Lester and R. Tombs, arXiv:2111.00616.
- [49] R. Tombs and C. G. Lester, *J. Instrum.* **17**, P08024 (2022).
- [50] Craigie *et al.* (to be published).
- [51] M. Abadi *et al.*, TensorFlow: Large-scale machine learning on heterogeneous systems (2015), software available from tensorflow.org.
- [52] F. Chollet *et al.*, Keras, <https://keras.io> (2015).
- [53] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, PyTorch: An imperative style, high-performance deep learning library (2017), https://proceedings.neurips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.
- [54] O. H. E. Philcox, M. J. König, S. Alexander, and D. N. Spergel, arXiv:2309.08653.
- [55] W. Wu, Z. Qi, and L. Fuxin, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 9621–9630, https://openaccess.thecvf.com/content_CVPR_2019/papers/Wu_PointConv_Deep_Convolutional_Networks_on_3D_Point_Clouds_CVPR_2019_paper.pdf.
- [56] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 652–660, https://openaccess.thecvf.com/content_cvpr_2017/papers/Qi_PointNet_Deep_Learning_CVPR_2017_paper.pdf.
- [57] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, *Adv. Neural Inf. Process. Syst.* **30** (2017), arXiv:1706.02413.
- [58] G. Qian, Y. Li, H. Peng, J. Mai, H. A. A. K. Hammoud, M. Elhoseiny, and B. Ghanem, arXiv:2206.04670.
- [59] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, *AI Open* **1**, 57 (2020).
- [60] S. Anagnostidis, A. Thomsen, T. Kacprzak, T. Tröster, L. Biggio, A. Refregier, and T. Hofmann, arXiv:2211.12346.
- [61] P. Villanueva-Domingo and F. Villaescusa-Navarro, *Astrophys. J.* **937**, 115 (2022).
- [62] T. L. Mäkinen, T. Charnock, P. Lemos, N. Porqueres, A. Heavens, and B. D. Wandelt, *Open J. Astrophys.* **5** (2022), [10.21105/astro.2207.05202](https://doi.org/10.21105/astro.2207.05202).
- [63] W. R. Coulton, O. H. E. Philcox, and F. Villaescusa-Navarro, *Phys. Rev. D* **109**, 023531 (2024).
- [64] Z. Slepian, arXiv:2306.05383.
- [65] P. Virtanen *et al.*, *Nat. Methods* **17**, 261 (2020).
- [66] C. R. Harris *et al.*, *Nature (London)* **585**, 357 (2020).
- [67] J. D. Hunter, *Comput. Sci. Eng.* **9**, 90 (2007).
- [68] A. F. Agarap, arXiv:1803.08375.
- [69] D. P. Kingma and J. Ba, arXiv:1412.6980.
- [70] A. Krizhevsky, I. Sutskever, and G. E. Hinton, *Commun. ACM* **60**, 84 (2017).